

# Dependence Maximizing Temporal Alignment via Squared-Loss Mutual Information

Makoto Yamada<sup>1</sup>, Leonid Sigal<sup>2</sup>, Michalis Raptis<sup>2</sup>, and Masashi Sugiyama<sup>1</sup>

<sup>1</sup>Tokyo Institute of Technology, 2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan

<sup>2</sup>Disney Research Pittsburgh, 4720 Forbes Ave., Pittsburgh, PA 15213

{yamada@sg. sugi@}cs.titech.ac.jp

March 8, 2013

## Abstract

The goal of *temporal alignment* is to establish time correspondence between two sequences, which has many applications in a variety of areas such as speech processing, bioinformatics, computer vision, and computer graphics. In this paper, we propose a novel temporal alignment method called *least-squares dynamic time warping* (LSDTW). LSDTW finds an alignment that maximizes statistical dependency between sequences, measured by a squared-loss variant of mutual information. The benefit of this novel information-theoretic formulation is that LSDTW can align sequences with different lengths, different dimensionality, high non-linearity, and non-Gaussianity in a computationally efficient manner. In addition, model parameters such as an initial alignment matrix can be systematically optimized by cross-validation. We demonstrate the usefulness of LSDTW through experiments on synthetic and real-world *Kinect* action recognition datasets.

## 1 Introduction

Temporal alignment of sequences is an important problem with many practical applications such as speech recognition [1, 2], activity recognition [3, 4], temporal segmentation [5], curve matching [6], chromatographic and micro-array data analysis [7], synthesis of human motion [8], and temporal alignment of human motion [9, 10].

*Dynamic time warping* (DTW) is a classical temporal alignment method that aligns two sequences by minimizing the pairwise squared Euclidean distance [1, 2]. An advantage of DTW is that the minimization can be efficiently carried out by *dynamic programming* (DP) [11]. However, due to the Euclidean formulation, DTW may not be able to find a good alignment when the characteristics of the two sequences are substantially different (e.g., sequences have different amplitudes). Moreover, DTW cannot handle sequences with different dimensions (e.g., image to audio alignment), which limits the range of applications significantly.

To overcome the weaknesses of DTW, *canonical time warping* (CTW) was introduced [9]. CTW performs sequence alignment in a common latent space found by canonical correlation analysis (CCA) [12]. Thus, CTW can naturally handle sequences with different dimensions. However, CTW can only deal with linear projections, and it is difficult to optimize model parameters such as the initial alignment matrix, the regularization parameter used in CCA, and the dimensionality of the common latent space.

To handle non-linearity, *dynamic manifold temporal warping* (DMTW) was recently proposed in [4]. DMTW first transforms original data onto a one-dimensional non-linear manifold and then finds an alignment on this manifold using DTW. Although DMTW is highly flexible by construction, its performance depends heavily on the choice of the non-linear transformation and, moreover, it implicitly assumes the smoothness of sequences. For this reason, DMTW has limited applicability.

In this paper, we propose a novel information-theoretic temporal alignment method based on statistical dependence maximization. Our method, which we call *least-squares dynamic time warping* (LSDTW), employs a squared-loss variant of mutual information called *squared-loss mutual information* (SMI) as a dependency

measure. SMI is estimated by the method of *least-squares mutual information* (LSMI) [13], which is consistent estimator achieving the optimal non-parametric convergence rate to the true SMI. An advantage of the proposed LSDTW over existing methods is that it can naturally deal with non-linearity and non-Gaussianity in data through SMI. Moreover, cross-validation (CV) with respect to the LSMI criterion is possible, which allows selection of model parameters such as the initial alignment matrix, the Gaussian kernel width, and the regularization parameter. Furthermore, the formulation of LSDTW is quite general and does not require strong assumptions on the topology of the latent manifold (e.g., smoothness). Thus, LSDTW is expected to perform well in a broader range of applications. Indeed, through experiments on synthetic and real-world *Kinect* action recognition tasks, LSDTW is shown to be a promising alternative to existing temporal alignment methods.

## 2 Dependence Maximizing Temporal Alignment via SMI

In this section, we first formulate the problem of *dependence maximizing temporal alignment* (DMTA) and then develop a DMTA method based on *squared-loss mutual information* (SMI) [13].

### 2.1 Formulation of Dependence Maximizing Temporal Alignment (DMTA)

Given two sequences represented by a set of samples (ordered in time),

$$\{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^{d_x}\}_{i=1}^{n_x} \text{ and } \{\mathbf{y}_j \mid \mathbf{y}_j \in \mathbb{R}^{d_y}\}_{j=1}^{n_y},$$

the goal of DMTA is to find a temporal alignment such that the statistical dependency between two sets of samples is maximized. Note that  $n_x$  and  $d_x$  can, in general, be different from  $n_y$  and  $d_y$ .

Let  $\boldsymbol{\pi}^x$  and  $\boldsymbol{\pi}^y$  be alignment functions over  $\{1, \dots, n_x\}$  and  $\{1, \dots, n_y\}$ , and let  $\boldsymbol{\Pi}$  be the corresponding alignment matrix:

$$\begin{aligned} \boldsymbol{\Pi} &:= [\boldsymbol{\pi}^x \ \boldsymbol{\pi}^y]^\top \in \mathbb{R}^{2 \times m}, \\ \boldsymbol{\pi}^x &:= [\pi_1^x, \dots, \pi_m^x]^\top \in \{1, \dots, n_x\}^{m \times 1}, \\ \boldsymbol{\pi}^y &:= [\pi_1^y, \dots, \pi_m^y]^\top \in \{1, \dots, n_y\}^{m \times 1}, \end{aligned}$$

where  $m$  is the number of indexes needed to align the sequences and  $^\top$  denotes the transpose.  $\boldsymbol{\Pi}$  needs to satisfy the following three additional constraints:

- **Boundary condition:**  $[\pi_1^x \ \pi_1^y]^\top = [1 \ 1]^\top$  and  $[\pi_m^x \ \pi_m^y]^\top = [n_x \ n_y]^\top$ .
- **Continuity condition:**  $0 \leq \pi_t^x - \pi_{t-1}^x \leq 1$  and  $0 \leq \pi_t^y - \pi_{t-1}^y \leq 1$ .
- **Monotonicity condition:**  $t_1 \geq t_2 \rightarrow \pi_{t_1}^x \geq \pi_{t_2}^x, \pi_{t_1}^y \geq \pi_{t_2}^y$ .

Let us denote the paired samples aligned by  $\boldsymbol{\pi}^x$  and  $\boldsymbol{\pi}^y$  as

$$Z(\boldsymbol{\Pi}) := \{(\mathbf{x}_{\pi_i^x}, \mathbf{y}_{\pi_i^y})\}_{i=1}^m.$$

Then, the optimal alignment, denoted by  $\boldsymbol{\Pi}^*$ , is defined as the maximum of a certain statistical dependence measure  $D$  between the two sets  $\{\mathbf{x}_i\}_{i=1}^{n_x}$  and  $\{\mathbf{y}_j\}_{j=1}^{n_y}$ :

$$\boldsymbol{\Pi}^* := \underset{\boldsymbol{\Pi}}{\operatorname{argmax}} D(Z(\boldsymbol{\Pi})).$$

### 2.2 Least-Squares Dynamic Time Warping (LSDTW)

A popular measure of statistical dependence is *mutual information* (MI) [14], and its estimation has been studied thoroughly [15, 16, 17, 18, 19]. However, these MI approximations are computationally expensive, mainly due to the non-linearity introduced by the “log” function. In this paper, we propose to use a squared-loss variant of MI called *squared-loss MI* (SMI), which results in a simple and computationally efficient estimation algorithm called *least-squares dynamic time warping* (LSDTW).

### 2.2.1 Overview

The optimization problem of LSDTW is defined as

$$\mathbf{\Pi}^* := \underset{\mathbf{\Pi}}{\operatorname{argmax}} \operatorname{SMI}(Z(\mathbf{\Pi})), \quad (1)$$

where SMI is defined and expressed as

$$\begin{aligned} \operatorname{SMI}(Z) &= \frac{1}{2} \iint \left( \frac{p_{xy}(\mathbf{x}, \mathbf{y})}{p_x(\mathbf{x})p_y(\mathbf{y})} - 1 \right)^2 p_x(\mathbf{x})p_y(\mathbf{y}) d\mathbf{x}d\mathbf{y}, \\ &= \frac{1}{2} \iint \left( \frac{p_{xy}(\mathbf{x}, \mathbf{y})}{p_x(\mathbf{x})p_y(\mathbf{y})} \right) p_{xy}(\mathbf{x}, \mathbf{y}) d\mathbf{x}d\mathbf{y} - \frac{1}{2}, \end{aligned} \quad (2)$$

where  $p_{xy}(\mathbf{x}, \mathbf{y})$  is the joint density of  $\mathbf{x}$  and  $\mathbf{y}$ , and  $p_x(\mathbf{x})$  and  $p_y(\mathbf{y})$  are the marginal densities of  $\mathbf{x}$  and  $\mathbf{y}$ , respectively. SMI is the *Pearson divergence* [20] from  $p_{xy}(\mathbf{x}, \mathbf{y})$  to  $p_x(\mathbf{x})p_y(\mathbf{y})$ , while the ordinary MI is the *Kullback-Leibler divergence* [21] from  $p_{xy}(\mathbf{x}, \mathbf{y})$  to  $p_x(\mathbf{x})p_y(\mathbf{y})$ . SMI is non-negative and is zero if and only if  $\mathbf{x}$  and  $\mathbf{y}$  are statistically independent, as the ordinary MI.

Based on Eq.(1), we develop the following iterative algorithm for estimating  $\mathbf{\Pi}$ :

- (i) **Initialization:** Initialize the alignment matrix  $\mathbf{\Pi}$ .
- (ii) **Dependence estimation:** For the current  $\mathbf{\Pi}$ , obtain an SMI estimator  $\widehat{\operatorname{SMI}}(Z(\mathbf{\Pi}))$ .
- (iii) **Dependence maximization:** Given an SMI estimator  $\widehat{\operatorname{SMI}}(Z(\mathbf{\Pi}))$ , obtain the maximum alignment  $\mathbf{\Pi}$ .
- (iv) **Convergence check:** The above (ii) and (iii) are repeated until  $\mathbf{\Pi}$  fulfills a convergence criterion.

### 2.2.2 Dependence Estimation

In the dependence estimation step, we utilize a non-parametric SMI estimator called *least-squares mutual information* (LSMI) [13], which was shown to possess a superior convergence property [22]. Here, we briefly review LSMI.

**Basic Idea:** The key idea of LSMI is to directly estimate the *density ratio* in Eq.(2) [23],

$$r(\mathbf{x}, \mathbf{y}) := \frac{p_{xy}(\mathbf{x}, \mathbf{y})}{p_x(\mathbf{x})p_y(\mathbf{y})},$$

from paired samples  $Z(\mathbf{\Pi}) = \{(\mathbf{x}_{\pi_i^x}, \mathbf{y}_{\pi_i^y})\}_{i=1}^m$  without going through density estimation of  $p_{xy}(\mathbf{x}, \mathbf{y})$ ,  $p_x(\mathbf{x})$ , and  $p_y(\mathbf{y})$ . Here, the density-ratio function  $r(\mathbf{x}, \mathbf{y})$  is directly modeled as

$$r_{\alpha}(\mathbf{x}, \mathbf{y}) = \sum_{\ell=1}^m \alpha_{\ell} K(\mathbf{x}, \mathbf{x}_{\pi_{\ell}^x}) L(\mathbf{y}, \mathbf{y}_{\pi_{\ell}^y}), \quad (3)$$

where  $K(\mathbf{x}, \mathbf{x}')$  and  $L(\mathbf{y}, \mathbf{y}')$  are kernel functions (e.g., Gaussian kernels) for  $\mathbf{x}$  and  $\mathbf{y}$ , respectively.

Then, the parameter  $\alpha = (\alpha_1, \dots, \alpha_m)^{\top}$  is learned so that the squared error to the true density ratio is minimized:

$$J_0(\alpha) := \frac{1}{2} \iint (r_{\alpha}(\mathbf{x}, \mathbf{y}) - r(\mathbf{x}, \mathbf{y}))^2 p_x(\mathbf{x})p_y(\mathbf{y}) d\mathbf{x}d\mathbf{y}.$$

After a few lines of calculation,  $J_0$  can be expressed as

$$J_0(\alpha) = J(\alpha) + \operatorname{SMI}(Z(\mathbf{\Pi})) + \frac{1}{2},$$

where

$$\begin{aligned} J(\boldsymbol{\alpha}) &:= \frac{1}{2} \boldsymbol{\alpha}^\top \mathbf{H}_\Pi \boldsymbol{\alpha} - \mathbf{h}_\Pi^\top \boldsymbol{\alpha}, \\ H_{\Pi, \ell, \ell'} &:= \iint K(\mathbf{x}, \mathbf{x}_{\pi_\ell^x}) L(\mathbf{y}, \mathbf{y}_{\pi_\ell^y}) K(\mathbf{x}, \mathbf{x}_{\pi_{\ell'}^x}) L(\mathbf{y}, \mathbf{y}_{\pi_{\ell'}^y}) p_x(\mathbf{x}) p_y(\mathbf{y}) d\mathbf{x} d\mathbf{y}, \\ h_{\Pi, \ell} &:= \iint K(\mathbf{x}, \mathbf{x}_{\pi_\ell^x}) L(\mathbf{y}, \mathbf{y}_{\pi_\ell^y}) p_{xy}(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}. \end{aligned}$$

Since  $\text{SMI}(Z(\Pi))$  is constant with respect to  $\boldsymbol{\alpha}$ , minimizing  $J_0$  is equivalent to minimizing  $J$ .

**Computing the Solution:** Approximating the expectations in  $\mathbf{H}_\Pi$  and  $\mathbf{h}_\Pi$  included in  $J$  by empirical averages, we have the following optimization problem:

$$\min_{\boldsymbol{\alpha}} \left[ \frac{1}{2} \boldsymbol{\alpha}^\top \widehat{\mathbf{H}}_\Pi \boldsymbol{\alpha} - \widehat{\mathbf{h}}_\Pi^\top \boldsymbol{\alpha} + \frac{\lambda}{2} \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \right], \quad (4)$$

where  $\lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha} / 2$  is the regularization term to avoid overfitting,  $\lambda$  ( $\geq 0$ ) is the regularization parameter, and

$$\begin{aligned} \widehat{H}_{\Pi, \ell, \ell'} &:= \frac{1}{m^2} \sum_{i, j=1}^m K(\mathbf{x}_{\pi_i^x}, \mathbf{x}_{\pi_\ell^x}) L(\mathbf{y}_{\pi_j^y}, \mathbf{y}_{\pi_\ell^y}) K(\mathbf{x}_{\pi_i^x}, \mathbf{x}_{\pi_{\ell'}^x}) L(\mathbf{y}_{\pi_j^y}, \mathbf{y}_{\pi_{\ell'}^y}), \\ \widehat{h}_{\Pi, \ell} &:= \frac{1}{m} \sum_{i=1}^m K(\mathbf{x}_{\pi_i^x}, \mathbf{x}_{\pi_\ell^x}) L(\mathbf{y}_{\pi_i^y}, \mathbf{y}_{\pi_\ell^y}). \end{aligned}$$

Differentiating Eq.(4) with respect to  $\boldsymbol{\alpha}$  and equating it to zero, we can obtain the optimal solution  $\widehat{\boldsymbol{\alpha}}_\Pi$  analytically as

$$\widehat{\boldsymbol{\alpha}}_\Pi = (\widehat{\mathbf{H}}_\Pi + \lambda \mathbf{I})^{-1} \widehat{\mathbf{h}}_\Pi, \quad (5)$$

where  $\mathbf{I}$  is the  $m \times m$  identity matrix. Note that, LSMI has time complexity  $O(m^3)$  due to the matrix inversion. However, when the number of training data is large, we can reduce the number of kernels in Eq.(3) to  $l$  ( $< m$ ) by sub-sampling. With this approximation, the inverse matrix in Eq.(5) can be computed with time complexity  $O(l^3)$ .

Finally, the following SMI estimator can be obtained by taking the empirical average of Eq.(2) as

$$\widehat{\text{SMI}}(Z(\Pi)) = \frac{1}{2m} \sum_{i=1}^m r_{\widehat{\boldsymbol{\alpha}}_\Pi}(\mathbf{x}_{\pi_i^x}, \mathbf{y}_{\pi_i^y}) - \frac{1}{2}. \quad (6)$$

**Model Selection:** Hyper-parameters included in the kernel functions and the regularization parameter can be optimized by cross-validation with respect to  $J$  [13], which is described below.

First, samples  $\mathcal{Z} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$  are divided into  $K$  disjoint subsets  $\{\mathcal{Z}_k\}_{k=1}^K$  of (approximately) the same size. Then, an estimator  $\widehat{\boldsymbol{\alpha}}_{\mathcal{Z}_k}$  is obtained using  $\mathcal{Z} \setminus \mathcal{Z}_k$  (i.e., all samples without  $\mathcal{Z}_k$ ), and the approximation error for hold-out samples  $\mathcal{Z}_k$  is computed as

$$J_{\mathcal{Z}_k}^{(K\text{-CV})} := \frac{1}{2} \widehat{\boldsymbol{\alpha}}_{\mathcal{Z}_k}^\top \widehat{\mathbf{H}}_{\mathcal{Z}_k} \widehat{\boldsymbol{\alpha}}_{\mathcal{Z}_k} - \widehat{\mathbf{h}}_{\mathcal{Z}_k}^\top \widehat{\boldsymbol{\alpha}}_{\mathcal{Z}_k},$$

where, for  $|\mathcal{Z}_k|$  being the number of samples in subset  $\mathcal{Z}_k$ ,

$$\begin{aligned} [\widehat{H}_{\mathcal{Z}_k}]_{\ell, \ell'} &:= \frac{1}{|\mathcal{Z}_k|^2} \sum_{\mathbf{x} \in \mathcal{Z}_k} \sum_{\mathbf{y} \in \mathcal{Z}_k} K(\mathbf{x}, \mathbf{x}_\ell) L(\mathbf{y}, \mathbf{y}_\ell) K(\mathbf{x}, \mathbf{x}_{\ell'}) L(\mathbf{y}, \mathbf{y}_{\ell'}), \\ [\widehat{h}_{\mathcal{Z}_k}]_\ell &:= \frac{1}{|\mathcal{Z}_k|} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{Z}_k} K(\mathbf{x}, \mathbf{x}_\ell) L(\mathbf{y}, \mathbf{y}_\ell). \end{aligned}$$

This procedure is repeated for  $k = 1, \dots, K$ , and its average  $J^{(K\text{-CV})}$  is taken as

$$J^{(K\text{-CV})} := \frac{1}{K} \sum_{k=1}^K J_{\mathcal{Z}_k}^{(K\text{-CV})}.$$

Finally, we compute  $J^{(K\text{-CV})}$  for all model candidates, and choose the one with minimum  $J^{(K\text{-CV})}$ .

### 2.3 Dependence Maximization

Based on the empirical estimate of SMI, the dependence maximization problem is given as

$$\max_{\Pi} \widehat{\text{SMI}}(Z(\Pi)).$$

We here provide a computationally efficient approximation algorithm based on *dynamic programming* (DP) [11].

Let us rewrite the empirical SMI, Eq.(6), as

$$\widehat{\text{SMI}}(Z(\Pi)) = \frac{1}{2m} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \delta(\pi_i^x, \pi_j^y) r_{\hat{\alpha}_\Pi}(\mathbf{x}_i, \mathbf{y}_j) - \frac{1}{2},$$

where

$$\delta(\pi_i^x, \pi_j^y) = \begin{cases} 1 & \text{if } \mathbf{x}_{\pi_i^x} \text{ and } \mathbf{y}_{\pi_j^y} \text{ are paired,} \\ 0 & \text{otherwise.} \end{cases}$$

Then, the solution is updated with the current  $\Pi^{\text{old}}$  as

$$\Pi^{\text{new}} = \arg\max_{\Pi} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \delta(\pi_i^x, \pi_j^y) r_{\hat{\alpha}_{\Pi^{\text{old}}}}(\mathbf{x}_i, \mathbf{y}_j). \quad (7)$$

This problem can be efficiently solved by DP with time complexity  $O(n_x n_y)$  (see Appendix). Note, however, that the solution to Eq.(7) does not always increase the empirical SMI, Eq.(6); we update the alignment matrix  $\Pi$  only if the SMI score increases after the update.

## 3 Related Methods

In this section, we review existing temporal alignment methods which are based on pairwise distance minimization (not dependence maximization) and point out their potential weaknesses.

### 3.1 Dynamic Time Warping (DTW)

The goal of *dynamic time warping* (DTW) is, given two sequences of the *same* dimensionality and the *different* number of samples,  $\{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^{n_x}$  and  $\{\mathbf{y}_j \mid \mathbf{y}_j \in \mathbb{R}^d\}_{j=1}^{n_y}$ , to find an alignment such that the sum of pairwise distances between two sets is minimized [1, 2]:

$$\min_{\pi^x, \pi^y} \sum_{i=1}^m \|\mathbf{x}_{\pi_i^x} - \mathbf{y}_{\pi_i^y}\|^2,$$

where  $m$  is the number of indices needed to align the sequences.  $\pi^x$  and  $\pi^y$  need to satisfy the boundary, continuity, and monotonicity conditions (see Section 2.1). The above DTW optimization problem can be efficiently solved by DP with time complexity  $O(n_x n_y)$ .

A potential weakness of DTW is that it cannot handle sequences with different dimensions such as image to audio alignment. Moreover, even when the dimensionality of sequences is the same, DTW may not be able to find a good alignment of sequences with different characteristics such as sequences with different amplitudes. These drawbacks highly limit the applicability of DTW.

### 3.2 Canonical Time Warping (CTW)

*Canonical time warping* (CTW) can align sequences with different dimensions in a common latent space [9, 10].

The CTW optimization problem is given as

$$\min_{\mathbf{W}_x, \mathbf{W}_y, \mathbf{V}_x, \mathbf{V}_y} \|\mathbf{V}_x^\top \mathbf{X} \mathbf{W}_x^\top - \mathbf{V}_y^\top \mathbf{Y} \mathbf{W}_y^\top\|_{\text{Frob}}^2, \quad (8)$$

where  $\|\cdot\|_{\text{Frob}}$  is the Frobenius norm,  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_{n_x}] \in \mathbb{R}^{d_x \times n_x}$ ,  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_{n_y}] \in \mathbb{R}^{d_y \times n_y}$ ,  $\mathbf{W}_x \in \{0, 1\}^{m \times n_x}$  and  $\mathbf{W}_y \in \{0, 1\}^{m \times n_y}$  are binary selection matrices that need to be estimated to align  $\mathbf{X}$  and  $\mathbf{Y}$ , and  $\mathbf{V}_x \in \mathbb{R}^{d_x \times b}$  and  $\mathbf{V}_y \in \mathbb{R}^{d_y \times b}$  ( $b \leq \min(d_x, d_y)$ ) are linear projection matrices of  $\mathbf{x}$  and  $\mathbf{y}$  onto a common latent space, respectively. The above optimization problem can be efficiently solved by alternately solving CCA and DTW, where the alignment matrix obtained using DTW is usually used as initialization (initial alignment matrix).

However, since CTW finds a common latent space using CCA, it can only deal with linear and Gaussian temporal alignment problems. Thus, CTW cannot properly deal with multi-modal and non-Gaussian data. Another limitation of CTW is that comparing the alignment quality over different model parameters is not straightforward. This is because, for different model parameters, a common latent space found by CCA is generally different and thus the metric of the pairwise distance Eq.(8) is also different. For this reason, a systematic model selection method for the regularization parameter, dimensionality of the common latent space, and the initial alignment matrix has not been developed so far, to the best of our knowledge.

### 3.3 Dynamic Manifold Temporal Warping (DMTW)

*Dynamic manifold temporal warping* (DMTW) is a non-linear extension of CTW [4].

The DMTW optimization problem is defined as

$$\min_{\mathbf{W}_x, \mathbf{W}_y, \mathcal{F}_x, \mathcal{F}_y} \|\mathcal{F}_x(\mathbf{X}) \mathbf{W}_x^\top - \mathcal{F}_y(\mathbf{Y}) \mathbf{W}_y^\top\|_{\text{Frob}}^2,$$

where  $\mathcal{F}_x : \mathbb{R}^{d_x \times n_x} \rightarrow \mathbb{R}^{b \times n_x}$  and  $\mathcal{F}_y : \mathbb{R}^{d_y \times n_y} \rightarrow \mathbb{R}^{b \times n_y}$  are non-linear mapping functions that map  $\mathbf{x}$  and  $\mathbf{y}$  to a common latent subspace. DMTW first maps  $\mathbf{X}$  and  $\mathbf{Y}$  to a one-dimensional smooth manifold (i.e.,  $b = 1$ ) by the *tensor voting* method [24] and then align sequences on the manifold.

DMTW highly depends on a specific non-linear transformations and requires the smooth manifold assumption. Thus, the usage of DMTW is limited to specific applications. On the other hand, CTW and LSDTW do not require the latter strong assumption and thus can be useful for a broader range of applications. This assertion will be experimentally validated in the next section.

## 4 Experiments

In this section, we experimentally evaluate our proposed LSDTW method on synthetic and real-world *Kinect* action recognition tasks.

### 4.1 Setup

In LSDTW, we use the Gaussian kernels:

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma_x^2}\right), \quad L(\mathbf{y}, \mathbf{y}') = \exp\left(-\frac{\|\mathbf{y} - \mathbf{y}'\|^2}{2\sigma_y^2}\right),$$

where  $\sigma_x$ ,  $\sigma_y$ , and  $\lambda$  are chosen by 3-fold CV from

$$(\sigma_x, \sigma_y) = c \times (m_x, m_y), \quad c = 2^{-1/2}, 1.8^{-1/2}, \dots, 0.2^{-1/2}, \quad \lambda = 10^{-1}, 10^{-2},$$

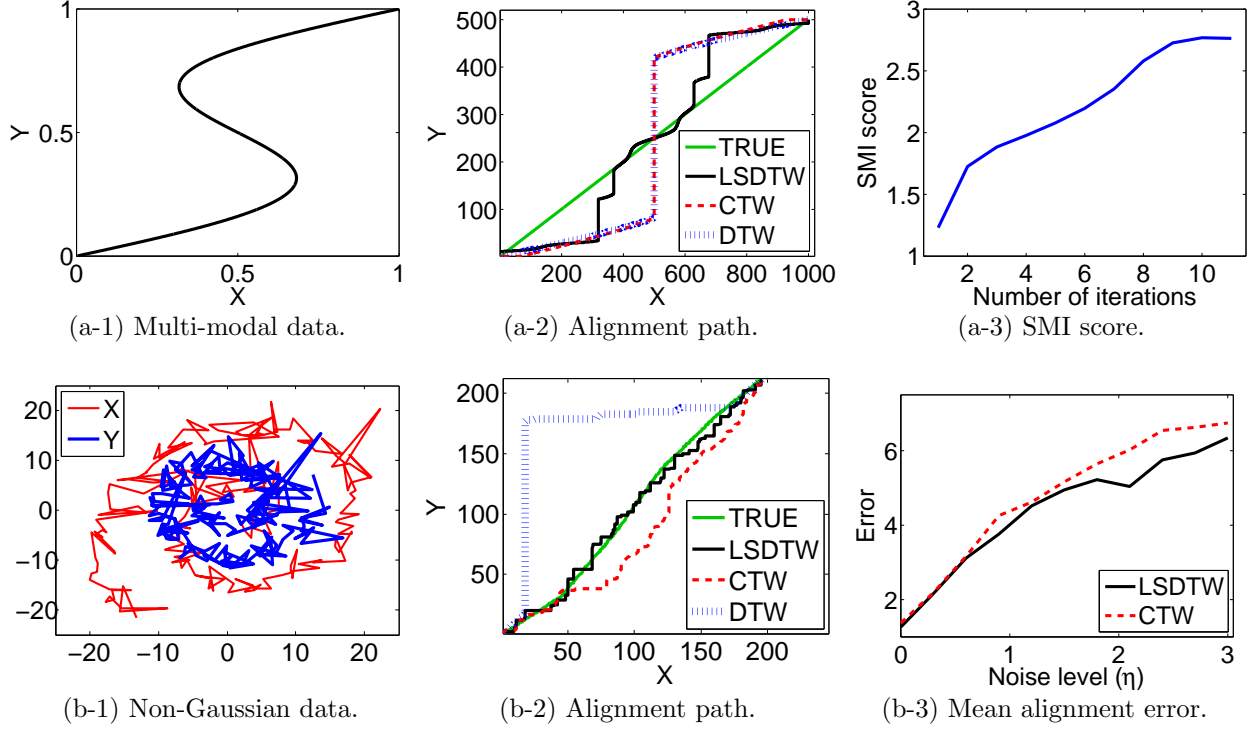


Figure 1: Results of synthetic experiments. (a-1) Multi-modal data. (a-2) Alignment paths. (a-3) SMI score as a function of the number of iterations. (b-1) A synthetic dataset with an additive exponential noise ( $\eta = 2.4$ ). (b-2) Alignment paths. Here, the alignment error of LSDTW, CTW, and DTW are 5.93, 26.28, and 87.69, respectively. (b-3) The mean alignment error over 100 runs as functions of noise level  $\eta$ .

and

$$m_x = 2^{-1/2} \text{median}(\{\|\mathbf{x}_i - \mathbf{x}_j\|\}_{i,j=1}^{n_x}), \quad m_y = 2^{-1/2} \text{median}(\{\|\mathbf{y}_i - \mathbf{y}_j\|\}_{i,j=1}^{n_y}).$$

Due to non-convex nature of the objective, setting a good initial alignment is an important issue for LSDTW. Here, from the alignment obtained using CTW and the simple uniform initialization,

$$\begin{aligned} \boldsymbol{\pi}^x &= [1, \lfloor 1 + n_x/m \rfloor, \lfloor 1 + 2n_x/m \rfloor, \dots, n_x]^\top \in \mathbb{R}^{m \times 1}, \\ \boldsymbol{\pi}^y &= [1, \lfloor 1 + n_y/m \rfloor, \lfloor 1 + 2n_y/m \rfloor, \dots, n_y]^\top \in \mathbb{R}^{m \times 1}, \end{aligned}$$

where  $m = \min(n_x, n_y)$ , we choose the one with the largest SMI score as the initial alignment for LSDTW.

We compare the performance of LSDTW with DTW and CTW. For CTW, we choose the dimensionality of CCA to preserve 90% of the total correlation, and we fix the regularization parameter at 0.01. We use the alignment given by DTW as the initial alignment of CTW.

To evaluate the alignment results, we use the following standard alignment error [10]:

$$Error = \frac{\text{dist}(\boldsymbol{\Pi}^*, \hat{\boldsymbol{\Pi}}) + \text{dist}(\hat{\boldsymbol{\Pi}}, \boldsymbol{\Pi}^*)}{m^* + \hat{m}}, \quad \text{dist}(\boldsymbol{\Pi}_1, \boldsymbol{\Pi}_2) = \sum_{i=1}^{m_1} \min(\{\|\boldsymbol{\pi}_1^{(i)} - \boldsymbol{\pi}_2^{(j)}\|\}_{j=1}^{m_2}),$$

where  $\boldsymbol{\Pi}^*$  and  $\hat{\boldsymbol{\Pi}}$  are true and estimated alignment matrices and  $\boldsymbol{\pi}_1^{(i)}, \boldsymbol{\pi}_2^{(j)} \in \mathbb{R}^{2 \times 1}$  are the  $i$ -th and  $j$ -th row of  $\boldsymbol{\Pi}_1$  and  $\boldsymbol{\Pi}_2$ , respectively.

## 4.2 Synthetic Dataset

First, we illustrate the behavior of the proposed LSDTW method for non-linear and non-Gaussian data using synthetic datasets.

**Non-linear (multi-modal) data<sup>1</sup>:**

$$\begin{aligned} x_i &= z_i + 0.4 \sin(2\pi z_i), \quad i = 1, \dots, 1000, \\ y_j &= z_{(j-1) \times 2 + 1}, \quad j = 1, \dots, 500, \end{aligned}$$

where  $z_i = i/1000$  (see Figure 1-(a-1)).

Figure 1-(a-2) shows the alignment path obtained by LSDTW, CTW, and DTW, respectively. In this experiment, we initialize LSDTW and CTW with the true alignment matrix and check whether those methods perform well. As can be observed, LSDTW can find a better alignment in the middle region (i.e., a multi-modal region) than DTW and CTW. This shows that LSDTW objective is much better than alternatives when it comes to multi-modal data. Figure 1-(a-3) depicts the SMI score with respect to the number of iterations in LSDTW, showing that LSDTW converges in 10 iterations.

**Non-Gaussian data:**

$$\mathbf{X} = \mathbf{U}_x^\top \mathbf{Z} \mathbf{M}_x^\top + \eta \mathbf{E}_x, \quad \mathbf{Y} = \mathbf{U}_y^\top \mathbf{Z} \mathbf{M}_y^\top + \eta \mathbf{E}_y,$$

where  $\mathbf{U}_x, \mathbf{U}_y \in \mathbb{R}^{2 \times 2}$  are randomly generated affine transformation matrices,  $\mathbf{Z} \in \mathbb{R}^{2 \times m}$  is a trajectory in two dimensions,  $\mathbf{M}_x \in \mathbb{R}^{n_x \times m}$  and  $\mathbf{M}_y \in \mathbb{R}^{n_y \times m}$  are randomly generated matrices for time warping,  $\mathbf{E}_x \in \mathbb{R}^{2 \times n_x}$  and  $\mathbf{E}_y \in \mathbb{R}^{2 \times n_y}$  are randomly generated additive exponential noise with rate parameter 1 (and its mean is adjusted to be zero), and  $\eta = \{0, 0.3, 0.6, \dots, 3.0\}$  is the noise level. Note that larger noise level  $\eta$  means stronger non-Gaussianity in the data.

Figures 1-(b-1) and (b-2) show an example of synthetic data with additive exponential noise ( $\eta = 2.4$ ) and corresponding alignment paths obtained by LSDTW, CTW, and DTW. As can be seen, only the proposed method can find a good alignment. Figure 1-(b-3) shows the mean alignment error over 100 runs, from which we can confirm that the proposed method tends to outperform the existing methods for larger noise levels.

## 4.3 Real-world Kinect Action Recognition Data

Next, we evaluate the proposed LSDTW method on the publicly available *Kinect* action recognition dataset<sup>2</sup> [25]. This dataset consists of the human skeleton data (15 joints) obtained using a *Kinect* sensor, and there are 16 subjects and 16 actions with 5 runs. Instead of using the raw skeleton data, we here use the 105-dimensional feature vector, where each element of the feature vector is the Euclidean distance between joint pairs.

In evaluation, we carry out *unsupervised* action recognition experiments and evaluate the performance of alignment methods by classification accuracy. More specifically, we first divide the action recognition dataset into two disjoint subsets: 8 subjects (#1-#8) with all actions for testing (in total 640 sequences) and the remaining subjects (#9-#16) with all actions for “training” database (in total 640 sequences). Then, we retrieve  $N = 10$  similar actions for each test action from the database by DTW, CTW, and LSDTW. Here, we use the pairwise Euclidean distance based on an estimated alignment to measure the similarity between sequences. Finally, if there is at least one correct action in the retrieved sequences, we regard the action to be correctly retrieved.

Figure 2 shows the mean classification accuracy as functions of the number of retrieved actions,  $N$ , where three different database sizes are tested. The graphs clearly show that LSDTW compares favorably with existing methods in terms of classification accuracy.

<sup>1</sup>A distribution of a multi-modal data has two or more modes.

<sup>2</sup>[www.cs.ucf.edu/~smasood/datasets/UCFKinect.zip](http://www.cs.ucf.edu/~smasood/datasets/UCFKinect.zip)



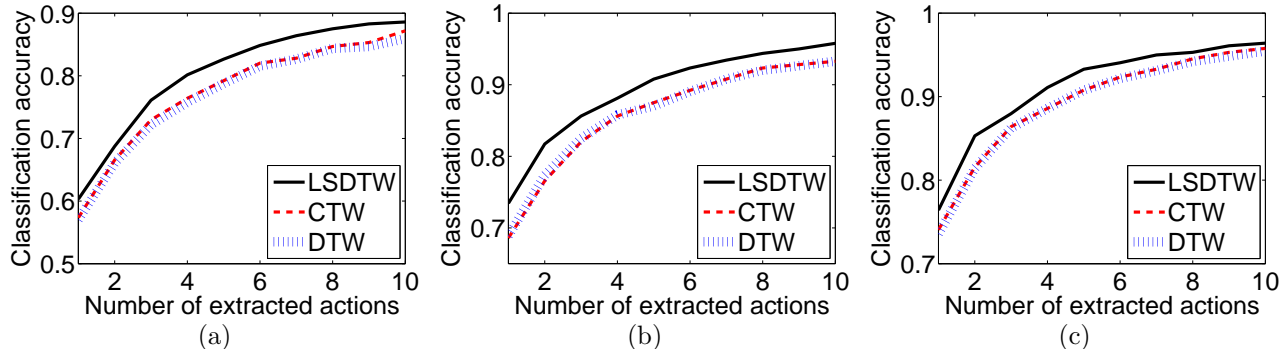


Figure 2: Mean classification accuracy with respect to the number of retrieved actions. (a) Only sequences of subject#9 are used as database. (b) Sequences of Subjects#9 – 12 are used as database. (c) Sequences of all subjects (#9 – 16) are used as database.

## 5 Conclusions

In this paper, we proposed a novel temporal alignment framework called the *dependence maximization temporal alignment* (DMTA) and developed a DMTA method called the *least-squares dynamic time warping* (LSDTW). LSDTW adopts *squared-loss mutual information* as a dependence measure, which is efficiently estimated by the method of *least-squares mutual information*. Notable advantages of LSDTW are that it can naturally deal with non-linear and non-Gaussian sequences and it can optimize model parameters such as the Gaussian kernel width and the regularization parameter by cross-validation. We applied the proposed method on the *Kinect* action recognition task, and experimentally showed that LSDTW is the promising alternative to the compared methods.

## Acknowledgments

We thank Prof. Fernando De la Torre, Mr. Feng Zhou, and Dr. Akisato Kimura for their valuable comments. MY acknowledges the JST PRESTO and PLIP programs, and MS acknowledges the JST PRESTO program and AOARD for financial support.

## References

- [1] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, (1):43–49, 1978.
- [2] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall Signal Processing Series, 1993.
- [3] I. M. Junejo, E. Dexter, I. Laptev, and P. Pérez. View-independent action recognition from temporal self-similarities. *IEEE TPAMI*, pages 172–185, 2011.
- [4] D. Gong and G. G. Medioni. Dynamic manifold warping for view invariant action recognition. In *ICCV*, 2011.
- [5] F. Zhou, F. De la Torre, and J. K. Hodgins. Aligned cluster analysis for temporal segmentation of human motion. In *FG*, 2008.
- [6] T. B. Sebastian, P. N. Klein, and B. B. Kimia. B.b.: On aligning curves. *IEEE TPAMI*, pages 116–124, 2003.

- [7] J. Listgarten, R. M. Neal, S. T. Roweis, and A. Emili. Multiple alignment of continuous time series. In *NIPS*, 2005.
- [8] E. Hsu, K. Pulli, and J. Popovi. Style translation for human motion. *ACM Transactions on Graphics*, 24:1082–1089, 2005.
- [9] F. Zhou and F. De la Torre. Canonical time warping for alignment of human behavior. In *NIPS*. 2009.
- [10] F. Zhou and F. De la Torre. Generalized time warping for multi-modal alignment of human motion. In *CVPR*, 2012.
- [11] R. Bellman. On the Theory of Dynamic Programming. In *Proceedings of the National Academy of Sciences*, volume 38, pages 716–719, 1952.
- [12] H. Harold. Relations Between Two Sets of Variates. *Biometrika*, 28(3/4):321–377, 1936.
- [13] T. Suzuki, M. Sugiyama, T. Kanamori, and J. Sese. Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, 10(S52), 2009.
- [14] C. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 1948.
- [15] Q. Wang, S. R. Kulkarni, and S. Verdú. Divergence estimation of continuous distributions based on data-dependent partitions. *IEEE Transactions on Information Theory*, 51(9):3064–3074, 2005.
- [16] M. Sugiyama, T. Suzuki, S. Nakajima, H. Kashima, P. von Büna, and M. Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.
- [17] F. Pérez-Cruz. Kullback-Leibler divergence estimation of continuous distributions. In *IEEE ISIT*, 2008.
- [18] J. Silva and S. S. Narayanan. Information divergence estimation based on data-dependent partitions. *Journal of Statistical Planning and Inference*, 140(11):3180–3198, 2010.
- [19] X. Nguyen, M. J. Wainwright, and M. I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.
- [20] K. Pearson. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50:157–175, 1900.
- [21] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.
- [22] T. Suzuki and M. Sugiyama. Sufficient dimension reduction via squared-loss mutual information estimation. In *AISTATS*, 2010.
- [23] M. Sugiyama, T. Suzuki, and T. Kanamori. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, Cambridge, UK, 2012.
- [24] P. Mordohai and G. G. Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *JMLR*, 11:411–450, 2010.
- [25] S. Z. Masood, A. Nagaraja, N. Khan, J. Zhu, and M. F. Tappen. Correcting cuboid corruption for action recognition in complex environment. In *ICCV Workshops*, 2011.

## Appendix

Given the empirical estimate of SMI computed at the dependence estimation step (Sect. 2.2.2), the dependence maximization problem is given as

$$\begin{aligned}
\max_{\mathbf{\Pi}} \widehat{\text{SMI}}(Z(\mathbf{\Pi})) &= \max_{\mathbf{\Pi}} \frac{1}{2m} \sum_{i=1}^m r_{\hat{\alpha}_{\mathbf{\Pi}_{\text{old}}}}(\mathbf{x}_{\pi_i^x}, \mathbf{y}_{\pi_i^y}) - \frac{1}{2} \\
&\stackrel{\text{Eq. (3)}}{=} \max_{\mathbf{\Pi}} \frac{1}{2m} \sum_{i=1}^m \sum_{\ell=1}^{m_{\text{old}}} \hat{\alpha}_{\ell} K(\mathbf{x}_{\pi_i^x}, \mathbf{x}_{\pi_{\ell}^x}^{\text{old}}) L(\mathbf{y}_{\pi_i^y}, \mathbf{y}_{\pi_{\ell}^y}^{\text{old}}) - \frac{1}{2} \\
&\approx \frac{1}{2(n_x + n_y)} \max_{\mathbf{\Pi}} \sum_{i=1}^m \sum_{\ell=1}^{m_{\text{old}}} \hat{\alpha}_{\ell} K(\mathbf{x}_{\pi_i^x}, \mathbf{x}_{\pi_{\ell}^x}^{\text{old}}) L(\mathbf{y}_{\pi_i^y}, \mathbf{y}_{\pi_{\ell}^y}^{\text{old}}) - \frac{1}{2}.
\end{aligned} \tag{9}$$

Based on the constraints on the alignment functions  $\mathbf{\Pi}$  described in Sect. 2.1, this optimal alignment can be computed by dynamic programming (DP) [11]. In order to verify this, we define the prefix sequences  $\mathbf{X}_n := \{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^{d_x}\}_{i=1}^n$  and  $\mathbf{Y}_{n'} := \{\mathbf{y}_j \mid \mathbf{y}_j \in \mathbb{R}^{d_y}\}_{j=1}^{n'}$ , with  $n \leq n_x$  and  $n' \leq n_y$ , and set  $A(n, n') := \widehat{\text{SMI}}(\mathbf{X}_n, \mathbf{Y}_{n'})$  denoting the optimal SMI for the aligned prefix sequences  $\mathbf{X}_n$  and  $\mathbf{Y}_{n'}$ .

Following the boundary conditions of the alignment functions, we have:

$$A(1, 1) = r_{\hat{\alpha}_{\mathbf{\Pi}_{\text{old}}}}(\mathbf{x}_1, \mathbf{y}_1). \tag{10}$$

Based on the continuity and monotonicity conditions, the DP-equation is given as

$$A(n, n') = \max\{A(n-1, n'-1), A(n-1, n'), A(n, n'-1)\} + r_{\hat{\alpha}_{\mathbf{\Pi}_{\text{old}}}}(\mathbf{x}_n, \mathbf{y}_{n'}), \tag{11}$$

for  $1 < n \leq n_x$  and  $1 < n' \leq n_y$ . Therefore, the optimal  $\widehat{\text{SMI}}(Z(\mathbf{\Pi})) = \frac{1}{2(n_x + n_y)} A(n_x, n_y) - \frac{1}{2}$  can be computed in  $O(n_x n_y)$ . Given the accumulated cost matrix  $A$ , we can compute the optimal alignment  $\mathbf{\Pi}$  using backtracking.